بررسی عملکرد الگوریتم‌های جستجوی فرااکتشافی و فراگیر جهت تحلیل رمز الگوریتم رمزنگاری SDES

میثم مرادی*[1] و مهدی عباسی[2]

[1]گروه کامپیوتر، دانشکده فنی و مهندسی، واحد علوم و تحقیقات همدان، دانشگاه آزاد اسلامی، همدان، ایران.
cn.m.moradi.co@gmail.com

[2]استادیارگروه مهندسی کامپیوتر، دانشکده فنی و مهندسی، دانشگاه بوعلی سینا، همدان، ایران.
abbasi@basu.ac.ir

چکیده

سالیان زیادی، تحلیل‌رمز به‌عنوان موضوعی جذاب در جهت به‌مخاطره‌انداختن امنیت و استحکام یک الگوریتم رمزنگاری مورد توجه قرار گرفته است. الگوریتم رمزنگاری SDES، یک الگوریتم رمزنگاری متقارن است که عملیات رمزنگاری را با استفاده از یک کلید، انجام می‌دهد. در دنیای رمزنگاری، الگوریتم‌های جستجوی متعددی جهت تحلیل رمز وجود دارد. در این پژوهش از الگوریتم جستجوی حمله فراگیر به‌عنوان یک الگوریتم جستجوی کامل، از الگوریتم ژنتیک به‌عنوان یک الگوریتم هوش تکاملی و از الگوریتم بهینه‌سازی توده ذرات به‌عنوان یک الگوریتم هوش جمعی استفاده شده است. هم‌راستا با این الگوریتم‌ها، یک الگوریتم ژنتیک پیشنهادی نیز با تنظیم و طراحی ابتکاری پارامترها و طراحی الگوریتمی جهت کشف کلید رمز نیز معرفی و سعی شده است، عملکرد الگوریتم‌های مختلف جهت تحلیل رمز الگوریتم رمزنگاری SDES مورد ارزیابی قرار بگیرد.

واژگان کلیدی: تحلیل‌رمز، استاندارد رمزگذاری دادهٔ ساده‌شده (SDES)، الگوریتم جستجوی حمله فراگیر، الگوریتم بهینه‌سازی توده ذرات، الگوریتم ژنتیک، کلید رمز.

۱- مقدمه

رمزنگاری[1] علمی است که براساس آن، اطلاعات و مفاهیم آشکار و قابل فهم برای همگان، طبق روالی برگشت‌پذیر به اطلاعات نامفهوم و گنگ تبدیل می‌شود. عمل تحلیل رمز[2] به مطالعه روش‌ها و اصولی می‌پردازد که براساس آن‌ها می‌توان بدون دراختیارداشتن کلید رمز، داده‌های رمزنگاری‌شده را از رمز خارج کرد، یا کلید رمز را به‌دست آورد. سامانهٔ رمزنگاری به دو روش رمزنگاری کلید عمومی[3] و رمزنگاری متقارن[4] تقسیم‌بندی می‌شود. در رمزنگاری کلید عمومی رمزگشایی و رمزگذاری با دو کلید متفاوت انجام، درحالی که در رمزنگاری کلید متقارن رمزگشایی و رمزگذاری با کلیدی مشابه انجام می‌شود. الگوریتم رمزنگاری DES به‌عنوان استاندارد رمزنگاری داده‌ها از سامانهٔ رمزنگاری کلید متقارن استفاده می‌کند [۱]. الگوریتم رمزنگاری SDES (استاندارد رمزگذاری دادهٔ ساده‌شده) یک نسخه از الگوریتم رمزنگاری DES است که در پژوهش‌ها مورد استفاده قرار می‌گیرد. در این پژوهش، کشف کلیدرمز الگوریتم SDES مورد بررسی قرار خواهد گرفت. تاکنون روش‌های مختلفی برای کشف کلیدرمز الگوریتم رمزنگاری SDES کشف شده‌اند. یکی از روش‌ها، آزمون همه حالت‌های مختلف کلید است که به جستجوی حمله فراگیر[5] معروف است. درصورتی‌که حالت‌های مختلف کلید کم باشد، می‌تواند روش مؤثری باشد. هر چه قدر طول کلید بزرگ‌تر باشد این عدد بزرگ‌تر و آزمون همه کلیدها بسیار زمان‌بر خواهد بود. برای رویارویی با این مشکل می‌توان از الگوریتم‌های جستجوی فرااکتشافی استفاده کرد. این الگوریتم‌ها به جای جستجوی تمام فضای حالت، جستجو را به بخشی از فضای کلید محدود می‌کنند و در زمان کمتری به جواب می‌رسند. الگوریتم‌های فرا اکتشافی به دو دسته هوش

[1] Cryptography
[2] Cryptanalysis
[3] Public Key Cryptosystem
[4] Symmetric Key System
[5] Brute Force

*نویسندهٔ مسؤول

تکاملی و هوش جمعی تقسیم می‌شوند [۲,۳]. الگوریتم ژنتیک یکی از الگوریتم‌های هوش تکاملی است که با الهام‌گرفتن از تکامل ژنتیکی موجودات زنده توسط آقای هلند ارائه شد [۳]. الگوریتم بهینه‌سازی توده ذرات[1] یکی از الگوریتم‌های هوش جمعی است که با الهام‌گرفتن از حرکت دسته‌جمعی پرندگان توسط ابراهات و جیمزکندی معرفی شد [۲]. در این پژوهش از الگوریتم جستجوی حمله فراگیر و الگوریتم فرااکتشافی جهت تحلیل رمز الگوریتم رمزنگاری SDES استفاده شده است.

ادامهٔ این پژوهش به شرح زیر سازمان‌دهی شده است:

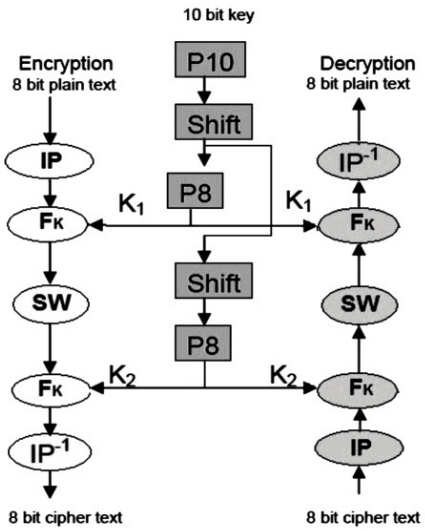
در بخش دوم به بررسی کارهای مرتبط پرداخته شده. در بخش سوم، مروری بر الگوریتم رمزنگاری SDES و در بخش چهارم، توصیف الگوریتم‌های جستجوی به‌کارگرفته‌شده در این پژوهش بررسی شده است. بخش پنجم طراحی آزمایش‌ها، در بخش ششم نتیجه و بحث به‌منظور ارزیابی الگوریتم‌های مختلف در تحلیل رمز الگوریتم رمزنگاری SDES و در بخش هفتم نتیجه‌گیری پژوهش با اشاره به پیشنهادهایی برای گسترش در آینده بررسی شده است.

۲- کارهای مرتبط

در سال‌های اخیر پژوهش‌های زیادی در حوزه تحلیل رمز انجام شده است. Sharma و همکاران [۲,۴]، شکستن الگوریتم SDES را با استفاده از الگوریتم بهینه‌سازی توده ذرات دودویی و ژنتیک بررسی کردند. Garg و همکاران [۳]، تحلیل رمز الگوریتم SDES را با استفاده از الگوریتم ژنتیک مورد مطالعه و پژوهش قرار دادند. Alani [۵]، جهت بهبود الگوریتم رمزنگاری DES در مقابل حمله فراگیر، تغییر طول کلید را پیشنهاد کرد. Alallayah و همکاران [۶]، شبکه عصبی را جهت تحلیل رمز DES مورد استفاده قرار دادند. Vimalathithan و همکاران [۹]، تحلیل‌رمز SDES را با استفاده از هوش تکاملی انجام دادند. Saroha و همکاران[۱۰]، یک گزارش از روش‌های مختلف تحلیل رمز ارائه دادند. Gopal و همکاران [۱۴]، یک روش بهینه‌شده جهت ارتباطات امن را با استفاده از الگوریتم DES معرفی کردند. Zodpe و همکاران [۱۵]، پیاده‌سازی سخت‌افزاری را برای تحلیل رمزها ارائه دادند. Jeswani و همکاران [۱۶]، یک الگوریتم بهینه‌سازی توده ذرات مبتنی بر تحلیل خطی را روی الگوریتم رمزنگاری AES به‌کار گرفتند. Bhateja [۱۷]، تحلیل رمز تفاضلی را با استفاده از روش‌های فرااکتشافی مورد مطالعه قرار داد. Salabat و همکاران [۱۸]، الگوریتم کلونی مورچه‌ها را برای تحلیل‌رمز DES4 به کار گرفتند. Dadhich و همکاران [۲۲]، رمزنگاری و تحلیل رمز را با استفاده از هوش جمعی و هوش تفاضلی مورد بررسی قرار دادند. Rajashekarappa و همکاران [۲۳]، تحلیل رمز SDES را با الگوریتم‌های ذوب فلزات و تابو مورد بررسی قرار دادند. Moradi و همکاران [۲۴]، شکستن الگوریتم SDES را با استفاده از الگوریتم بهینه‌سازی توده ذرات بهینه‌شده انجام دادند. Zoubir و همکاران [۲۵]، جهت تحلیل رمز الگوریتم رمزنگاری RSA از الگوریتم ژنتیک استفاده کردند. Kendhe و همکاران [۲۷]، یک گزارش از روی روش‌های مختلف تحلیل رمز ارائه دادند. در کارهای پیشین، تحلیل رمز روی الگوریتم‌های رمزنگاری با استفاده از روش‌های مختلفی انجام شده است. الگوریتم‌های فرااکتشافی روشی جدیدی در حوزه هوش مصنوعی است که در سال‌های اخیر بر روی تحلیل رمز الگوریتم‌های رمزنگاری به‌کار گرفته شده است. در این پژوهش سعی شده است که مطالعه‌ای روی الگوریتم‌های فرااکتشافی و فراگیر صورت گیرد و عملکرد این الگوریتم‌ها را در دو معیار کشف کلیدرمز و زمان کشف کلیدرمز الگوریتم رمزنگاری SDES مورد ارزیابی قرار گیرد.

۳- الگوریتم رمزنگاری SDES

این الگوریتم نسخه ساده‌شدهٔ DES است که در پژوهش‌های مختلف مورد استفاده قرار گرفته است. در این الگوریتم هشت بیت داده اصلی را با ده بیت کلیدرمز به عنوان ورودی گرفته و به هشت بیت دادهٔ رمزشده تبدیل می‌کند. الگوریتم رمزگشایی SDES به‌صورت معکوس رمزگذاری است [۳]. در زیر در مورد کلید رمز و همچنین توابع استفاده‌شده در این الگوریتم بحث شده که شمای کلی الگوریتم رمزنگاری SDES در شکل (۱) آمده است.



(شکل-۱): نمودار رمزنگاری [۳]

[1] Particle swarm optimization

۳-۱- تولید کلید فرعی از کلیدرمز

از کلید رمز ده‌بیتی دو کلید فرعی هشت‌بیتی استخراج شده است. در ابتدا می‌بایست کلید رمز به‌صورت P۱۰= [۳ ۵ ۲ ۷ ۴ ۱۰ ۱ ۹ ۸ ۶] جایگشت شود؛ سپس یک شیفت عملیاتی به چپ داده شده که خروجی شیفت عملیاتی به چپ به‌صورت P8= [۶ ۳ ۷ ۴ ۸ ۵ ۱۰ ۹] جایگشت (هشت بیتی) شده است و نخستین کلید فرعی تولید می‌شود. خروجی شیفت عملیاتی در این مرحله دوباره شیفت عملیاتی به چپ روی آن انجام می‌شود(دو شیفت عملیاتی) و با جایگشت هشت‌بیتی مانند P8 دومین کلید فرعی هم تولید می‌شود [۴].

۳-۲- فرایند رمزنگاری

فرایند رمزنگاری از مراحل زیر تشکیل شده است:

۱. جایگشت اولیه و نهایی[1]: ورودی الگوریتم هشت بیت دادهٔ رمزشده را با استفاده از تابع IP = [۲ ۶ ۳ ۱ ۴ ۸ ۵ ۷] جایگشت می‌دهد، در پایان کار، معکوس جایگشت داده‌شده به‌صورت تابع IP^{-1} = [۴ ۱ ۳ ۵ ۷ ۲ ۸ ۶] اعمال می‌شود.

۲. تابع کلید[2]: این تابع، تابع پیچیده‌ای در SDES است که شامل ترکیبی از جایگشت و جانشینی تابع است [۴]. نیمه چپ و راست به‌صورت زیر مقداردهی می‌شود که تابع F در نیمهٔ راست، در بخش دوم رابطه (۱) آمده است [۱]:

$$\begin{cases} L_{i-1} = R_{i-1} \\ R_{i-1} = L_{i-1} \oplus F(R_{i-1}, K_i) \end{cases} \quad (1)$$

که L و R چهار بیت در بخش چپ و راست ورودی هستند. XOR، عملیات OR انحصاری است و Key هم کلید فرعی است [۱]. محاسبه F(R,Key) در زیر آمده است:

۱- اضافه‌کردن توسعه – جایگشت چهاربیتی که به‌صورت E/P = [۴ ۱ ۲ ۳ ۲ ۳ ۴ ۱] اعمال می‌شود:

۲- اضافه‌کردن هشت بیت کلید فرعی و XOR کردن آن با خروجی مرحلهٔ یک:

۳- چهار بیت سمت چپ درجعبه جایگشت S_0[3] و چهار بیت سمت راست در جعبه جایگشت S_1 قرار گرفته است:

۴- خروجی به‌صورت P۴ = [۲ ۴ ۳ ۱] جایگشت می‌شود.

جعبه جایگشت به‌صورت زیر کار می‌کند:

نخستین و چهارم بیت به‌عنوان سطر، دومین و سومین بیت به عنوان ستون جعبه جایگشت در نظر گرفته می‌شود و روی سطر و ستون در نهایت چهار بیت خروجی به‌دست می‌آید. جعبه جایگشت تعریف‌شده به‌صورت جدول (۱) است:

(جدول-۱): جعبه جایگشت مورد استفاده

S_0				S_1			
۱	۰	۳	۲	۰	۱	۲	۳
۳	۲	۱	۰	۲	۰	۱	۳
۰	۲	۱	۳	۳	۰	۱	۰
۳	۱	۳	۲	۲	۱	۰	۳

تابع سوییچ[4] چهار بیت سمت چپ و راست را با هم عوض می‌کند و برای دور دوم توابع E/P، S0، S1 و P4 با استفاده از کلید فرعی دوم، عملیاتی مشابه دور نخست انجام می‌شود و الگوریتم رمزگذاری با دو دور خاتمه می‌یابد [۴].

۳-۳- الگوریتم رمزگشایی

رمزگشایی، معکوس الگوریتم رمزگذاری است. هشت بیت دادهٔ رمزشده با ده بیت کلید رمز، به هشت بیت داده اصلی تبدیل می‌شود. بلوک رمزگشایی مثل بلوک رمزگذاری است، با این تفاوت که جای کلیدهای فرعی یک و دو عوض و تابع FK معکوس می‌شود.

۳-۴- رمزنگاری و رمزگشایی متون

جهت رمزنگاری متن‌ها، از بخش رمزنگاری الگوریتم SDES، هر حرف در یک بلوک متنی در ورودی الگوریتم قرار گرفته و عملیات رمزنگاری بر روی آن انجام و جهت رمزگشایی متن‌ها، از بخش رمزگشایی الگوریتم SDES، برای هر حرف در یک بلوک متنی استفاده و عملیات رمزگشایی انجام می‌شود.

۴- الگوریتم‌های جستجو

در حل مسائل کاربردی نیاز به جستجو امری غیر قابل اجتناب و در عین حال دشوار است. به همین جهت تعداد زیادی از الگوریتم‌های جستجو با فلسفه‌ها و دامنهٔ استفاده متفاوت به‌وجود آمده‌اند. این الگوریتم‌های جستجو را می‌توان به دو دسته کلی جستجوهای کامل و جستجوهای مکاشفه‌ای تقسیم کرد. تفاوت اساسی بین الگوریتم‌های این دو دسته به این صورت است که در جستجوهای کامل، تمام فضای جستجو به‌طور کامل مورد جستجو و ارزیابی قرار می‌گیرد تا جواب مورد نظر یافته شود؛ درحالی‌که در جستجوهای مکاشفه‌ای تنها بخشی از فضا که احتمال یافتن جواب در آن بیشتر است،

[1] Intial Permutation
[2] Function Key
[3] Sbox
[4] Switch

مورد توجه قرار می‌گیرد که در ادامه به معرفی برخی از این الگوریتم‌ها پرداخته شده است.

۴-۱- الگوریتم جستجوی فراگیر (Brute Force)

الگوریتم جستجوی حمله فراگیر به‌عنوان یک الگوریتم جستجوی کامل است؛ در این الگوریتم، همه حالت‌های مختلف کلید آزمون می‌شود و در صورت وجود جواب، آن را پیدا می‌کند؛ به همین دلیل به جستجوی حمله فراگیر معروف است. درصورتی‌که حالت‌های مختلف کلید کم باشد، می‌تواند روش مؤثری باشد. هر چه قدر طول کلید بزرگ‌تر باشد؛ این عدد بزرگ‌تر و آزمون همه کلیدها بسیار زمان‌بر خواهد بود. الگوریتم جستجوی فراگیر یک معیار جامع جهت شناخت روش‌های مختلف تحلیل رمز شناخته می‌شود؛ به این معنی که هر روشی که سریع‌تر از روش جستجوی حمله فراگیر بتواند رمز را بازگشایی کند، به‌عنوان یک روش تحلیل رمز می‌تواند شناخته شود.

۴-۲- الگوریتم ژنتیک (Gentic Algorithm)

الگوریتم ژنتیک یکی از الگوریتم‌های فرا اکتشافی جهت حل مسائل بهینه‌سازی است که ریشه آن از قانون بقای اصلح سرچشمه می‌گیرد. درواقع می‌توان گفت که این الگوریتم یک شبیه‌سازی مجازی از نظریه تکامل تدریجی داروین است [۳]. این الگوریتم کار خود را با یک جمعیت اولیه آغاز کرده و در هر تکرار محاسباتی (نسل) روی جمعیتی از کروموزوم‌ها عمل کرده و تغییرات تصادفی روی مجموعه کروموزوم‌ها از طریق اعمال عمل‌گرهای ژنتیکی (جهش و ترکیب) انجام می‌دهد. پس از اعمال این عمل‌گرها دنباله کروموزوم‌ها از نظر عملکرد بر اساس تابع هدف ارزیابی شده و انتخاب برای نسل‌های بعدی بر مبنای این تابع ارزیابی انجام می‌شود تا نسل‌به‌نسل به سمت تکامل پیش رود و به جواب موردنظر برسد. به‌دلیل روال تکاملی که در ساختار این الگوریتم وجود دارد به‌عنوان الگوریتم هوش تکاملی شناخته می‌شود. این الگوریتم عمل‌گرهای مهمی دارد که در الگوریتم ژنتیک پیشنهادی بیشتر به معرفی این پارامترها پرداخته می‌شود.

۴-۳- الگوریتم بهینه‌سازی توده ذرات (PSO)

PSO یکی از الگوریتم فرااکتشافی است که توسط راسل ابرهارت دانشمند علوم رایانه و جیمزکندی روان‌شناس مسائل اجتماعی معرفی شد. این الگوریتم یک روش بهینه‌سازی مبتنی بر قواعد احتمال است و از رفتار اجتماعی پرندگان یا ماهی‌ها در پیداکردن غذا الهام گرفته شده است [۲]. پرندگان تنها با تنظیم حرکت فیزیکی خود و اجتناب از تصادم به دنبال غذا می‌گردند و به‌طور تئوری هر پرنده به‌عنوان یکی از اعضای گروه از تجربه قبلی خود و یافته‌های سایر اعضا برای یافتن غذا بهره می‌برد. این مشارکت یک برتری قطعی برای یافتن غذا است. پایهٔ اصلی نظریه PSO همین تسهیم اطلاعات بین اعضا در یک گروه است؛ به همین دلیل به‌عنوان الگوریتم هوش جمعی شناخته می‌شود.

در هر مرحله از حرکت جمعیت هر ذره با دو مقدار بهترین، به‌روز می‌شود، نخستین مقدار بهترین جواب از لحاظ شایستگی است که تاکنون برای هر ذره به‌طور جداگانه به‌دست آمده است و Pbest نامیده می‌شود. مقدار بهترین دیگر، بهترین مقداری است که تاکنون توسط تمام ذره‌ها در میان جمعیت به‌دست آمده است. این مقدار بهترین کلی است و Gbest نامیده می‌شود. مقادیرPbest و Gbest براساس یک تابع ارزیابی انتخاب می‌شود. این تابع برای مسائل مختلف متفاوت است. بعد از یافتن دو مقدار Pbest و Gbest هر ذره به صورت چند بعدی با دو مقدار Xid و Vid که به‌ترتیب معرف وضعیت مکانی و سرعت مربوط به بعد dام از iاُمین ذره است سرعت و مکان جدید خود را به روز می‌کند. این فرآیند با استفاده از رابطه‌های (۲و۳) زیر صورت می‌گیرد:

$$Vi[t+1] = W * Vi[t] + C_1 * RAND() * (P_{Best}[t] - Xi[t]) + C_2 * RAND() * (G_{Best}[t] - Xi[t]) \quad (۲)$$

$$Xi[t+1] = Xi[t] + Vi[t+1] \quad (۳)$$

در رابطه‌های بالا w وزن اینرسی، c1 و c2 عوامل یادگیری که به آن‌ها ضرایب شتاب نیز گفته می‌شود و (Rand) عددی تصادفی بین صفر و یک است. مقدار این پارامترها در حل مسائل مختلف متفاوت است. مقدار این پارامترها در همگرایی مسئله بسیار موثر هستند. در [۲] مقدار بین صفر تا ۰/۹۹برای W و مقدار۲=C1=C2 پیشنهاد شده است . به هر حال انتخاب مقدار این پارامترها به نوع مسأله بستگی دارد.

۴-۴- الگوریتم ژنتیک پیشنهادی:

ایده اصلی الگوریتم ژنتیک توسط جان هلند مطرح شد [۳]. الگوریتم ژنتیک برای حل یک مسأله، مجموعه بسیار بزرگی از راه‌حل‌های ممکن را تولید می‌کند. هر یک از این راه‌حل‌ها با استفاده از یک تابع برازندگی مورد ارزیابی قرار می‌گیرد؛ آن‌گاه

تعدادی از بهترین راه‌حل‌ها باعث تولید راه‌حل‌های جدیدی می‌شوند که این کار باعث تکامل راه‌حل‌ها می‌شود. بدین ترتیب فضای جستجو در جهتی تکامل پیدا می‌کند که به راه‌حل مطلوب برسد. اساس کار این الگوریتم قانون تکامل داروین است که در این پژوهش براساس این قانون، کلیدهای ضعیف‌تر (با برازندگی کمتر) از بین می‌روند وکلیدهای قوی‌تر (با برازندگی بیشتر) باقی می‌مانند. درواقع تکامل فرایندی است که روی کلیدها صورت می‌گیرد و اساس قانون انتخاب طبیعی برای بقا به صورتی است که هر چه امکان تطبیق کلید بیشتر باشد، بقای کلید امکان پذیرتر است و برای بقای نسل، با استفاده از پارامترهایی جمعیت جدید کلیدها تولید می‌شود. در این پژوهش سعی شده است که با تنظیم پارامترهایی نظیر انتخاب، جمعیت، تعداد نسل و طراحی ابتکاری پارامترهای نظیر برش، جهش، تابع برازندگی در الگوریتم ژنتیک و با پیشنهاد الگوریتمی جهت کشف کلید رمز، به تحلیل رمز الگوریتم رمزنگاری SDES پرداخته شده باشد.

۴-۴-۱- طراحی الگوریتم و روندنما جهت کشف کلید رمز الگوریتم رمزنگاری SDES

الگوریتمی که در الگوریتم ژنتیک پیشنهادی جهت کشف کلیدرمز الگوریتم SDES در نظر گرفته شده به شرح زیر است:

۱- ورودی الگوریتم، متن رمزشده است؛

۲- تولید تصادفی جمعیت اولیه به‌عنوان کلیدهای‌رمز؛

۳- ارزیابی هرکلید با معکوس الگوریتم SDES و متن رمزشده برای به‌دست آوردن متن‌رمز؛

۴- محاسبه برازندگی برای هر کلید با توجه به تابع برازندگی؛

۵- ارزیابی شرط که در صورت برقراری شرط (پایان نسل) الگوریتم پایان می‌یابد در غیر این صورت به گام شش می‌رود.

۶- نسل جدید به‌صورت زیر تولید می‌شود:

۶-۱- انتخاب والدها با بالاترین برازندگی(نخبه‌گرایی) برای استفاده درنسل جدید؛

۶-۲- هر جفت والد بر اساس برش ستاره‌ای فرزندانی تولید می‌کند؛

۶-۳- هر والد براساس عمل‌گر جهش سه‌نقطه‌ای- تصادفی فرزندانی تولید می‌کند؛

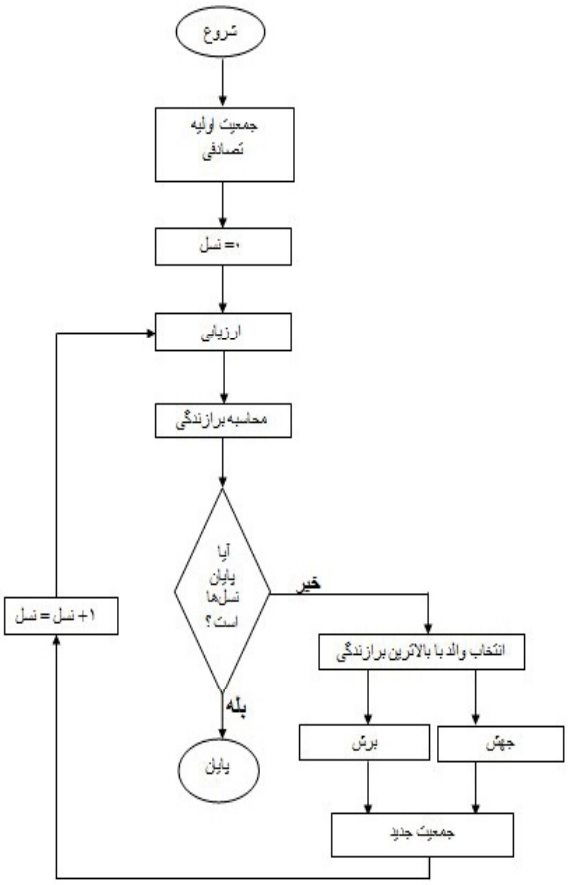
۶-۴- همه فرزندان تولیدشده توسط عمل‌گرهای الگوریتم ژنتیک در نسل جدید قرار می‌گیرند.

۶-۵- رفتن به مرحله ۳.

[1] Selection

روندنمای الگوریتم ژنتیک پیشنهادی در شکل (۲) آمده است.

یکی از عامل‌های تأثیرگذار در عملکرد الگوریتم ژنتیک پارامترهای آن است. از الگوریتم‌های ژنتیک در حوزه‌های مختلف استفاده می‌شود. عملکرد الگوریتم ژنتیک با طراحی مناسب و دقیق پارامترها(عمل‌گرها) در هرحوزه کاری، می‌تواند عملکرد متفاوتی از خود نشان دهد. در این پژوهش سعی شده که براساس آزمون‌های مختلف در جهت کشف کلید رمز الگوریتم رمزنگاری SDES، در الگوریتم ژنتیک پارامترهایی تنظیم و بازطراحی شوند که در زیر به توصیف آن پرداخته می‌شود.



(شکل-۲): روندنمای الگوریتم ژنتیک پیشنهادی

۴-۴-۲- تنظیم و طراحی ابتکاری پارامترها براساس کشف کلیدرمز الگوریتم رمزنگاری SDES

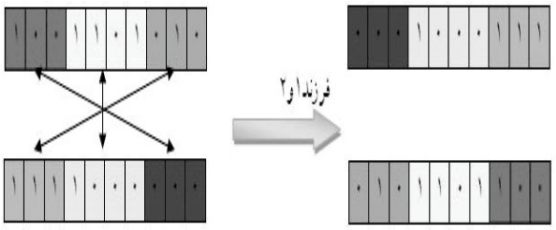
۴-۴-۲-۱- پارامتر انتخاب

در این پژوهش از عمل‌گر انتخاب[1] جهت تولید کلیدها در نسل جدید استفاده می‌شود. عمل‌گر انتخاب می‌تواند به‌صورت بهترین- بهترین، بهترین- بدترین و... صورت بپذیرد. در این پژوهش براساس آزمون‌های انجام‌شده از ایده نخبه‌گرایی استفاده شده است. در ایده نخبه‌گرایی بهترین کلیدها هر نسل

براساس بالاترین برازندگی زنده می‌مانند و در نسل‌های بعد حضور می‌یابند و کلیدها با برازندگی پایین از بین می‌روند. دلیل استفاده از ایده نخبه‌گرایی براساس آزمون‌های انجام‌شده این بوده است که هر چقدر که برازندگی کلیدها بالاتر باشد، به‌منزلهٔ این است که متن کشف‌شده از لحاظ بیتی مشابهت بیشتری در مقایسه با متن اصلی دارا است و اصول این الگوریتم بر پیش‌روی به سوی بهینگی و تکامل است و براساس این اصول، در هر نسل به کشف کلید رمز نزدیک و نزدیک‌تر می‌شود و در کاهش زمان جستجو هم مؤثر است.

۴-۴-۲-۲- پارامتر برش

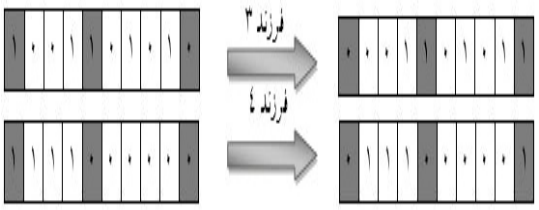
مهم‌ترین عمل‌گر در الگوریتم ژنتیک عمل‌گر برش[1] است در این عمل‌گر کلیدهایی که به‌عنوان والد در قسمت انتخاب در نظر گرفته می‌شوند، ضمن حفظ ساختار بیت‌های کلیدهای والد، مکان بیت‌های آن‌ها با هم مبادله می‌شوند. عمل‌گر برش به‌صورت یک‌نقطه‌ای، چندنقطه‌ای، حلقوی و غیره صورت می‌گیرد. در این پژوهش، از عمل‌گر برش جهت تولید کلیدها (فرزندان) در بقای نسل‌ها استفاده شده است و براساس آزمون‌های انجام‌شده برش ستاره‌ای جهت کشف کلید رمز طراحی شده است. دلیل استفاده از برش ستاره‌ای بر اساس آزمون‌ها این بوده است که در بهینه محلی[2] گیر نیافتد، پدیده‌ای که باعث می‌شود در چند نسل کلیدهای تکراری تولید شود و کلید رمز کشف نشود و در ضمن زمان جستجو نیز بالا می‌رود. برش ستاره‌ای در شکل (۳) آمده است.



(شکل-۳): برش ستاره‌ای

۴-۴-۲-۳- پارامتر جهش

جهش[3] عمل‌گر دیگری است که جهت تولید کلیدها (فرزندان) در نسل جدید استفاده می‌شود. عمل جهش در یک بیت با متمم‌ساختن آن بیت انجام می‌شود، به این صورت که اگر بیت مورد نظر صفر بود به یک و بر عکس انجام می‌شود. هر بیت با احتمال جهش، جهش می‌یابد. این عمل‌گر به‌صورت یک‌نقطه‌ای، دونقطه‌ای و ... صورت می‌پذیرد. در این پژوهش، عمل‌گر جهش با احتمال ۰/۳ (سه نقطه) و به‌صورت تصادفی طراحی شده است. دلیل استفاده از جهش سه‌نقطه‌ای- تصادفی بر اساس آزمون‌های انجام‌شده این بوده است که با تغییر سه بیت از کلید به‌صورت تصادفی، کلیدهای متنوعی تولید می‌شود و در نتیجه در بهینه محلی گیر نمی‌افتد و در کاهش زمان هم مؤثر است. جهش سه‌نقطه‌ای- تصادفی در شکل (۴) آمده است.



(شکل-۴): جهش سه نقطه ای- تصادفی

۴-۴-۲-۴- تابع برازندگی

رابطه (۲)، یک تابع برازندگی[4] برای تعیین برازندگی عمومی کلید فرضی مناسب و زبان الفبایی برای انگلیسی ([A-Z]) است. K و D به‌ترتیب آمار زبان شناخته شده و آمار پیام رمزگشایی شده است. u، b و t به‌ترتیب تک‌حرفی، دوحرفی و سه‌حرفی آماری هستند. α ، β و ϒ ضریب وزنی تخصیص داده‌شده به هر سه حالت حرفی است که α + β + ϒ = ۱ است [۲]. در بیشتر پژوهش‌ها به‌دلیل پیچیدگی حالت‌های مختلف حروف در رمزگشایی و افزایش زمان جستجو از حالت تک‌حرفی استفاده شده است. در این پژوهش، با استفاده از همین تابع برازندگی و بررسی زمان جستجو، هر سه حالت حرفی پیاده‌سازی شده است (Λ, ΛS, THE, ...).

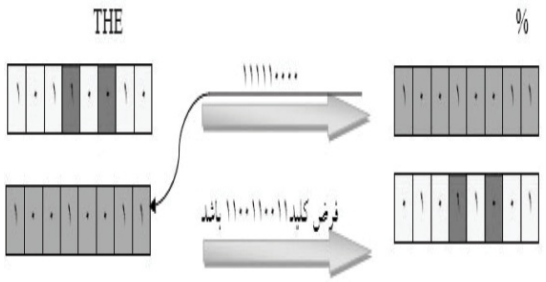
$$C_K = \alpha \Sigma(i\varepsilon\tilde{A})|k(i)u - D(i)u| + \beta \Sigma(i,j\varepsilon\tilde{A})|k(i,j)b - D(i,j)b| + \gamma \Sigma(i,j,k\varepsilon\tilde{A})|k(i,j,k)t - D(i,j,k)t| \quad (۴)$$

۴-۴-۲-۵ محاسبه برازندگی

محاسبه برازندگی کلید، از اِعمال تبدیل مناسب بر روی تابع برازندگی یعنی تابعی که قرار است بهینه شود به‌دست می‌آید. محاسبه برازندگی هر کلید براساس رمزگشایی متن رمزشده برای به‌دست‌آوردن متن اصلی طراحی شده است. به‌عنوان مثال در شکل (۵) حرف THE با کلید ۱۱۱۱۱۰۰۰۰۰ به حرف ٪ رمز و سپس هر کلیدی (فرض ۱۱۰۰۱۱۰۰۱۱ در این پژوهش) را با حرف ٪ که معادل با هشت بیت به‌عنوان ورودی است، رمز می‌شود و یک حرف با هشت بیت به عنوان

[1] Crossover
[2] Local Optimum
[3] Mutation
[4] Fittness Function

خروجی به‌دست می‌آید؛ سپس تعداد بیت‌های آن حرف با هشت بیت به عنوان خروجی را با حرف THE با هشت بیت به‌عنوان ورودی مقایسه می‌شود، تعداد مکان‌های که دارای بیت‌هایی یکسان هستند به‌عنوان برازندگی محاسبه می‌شود. در شکل (۵) برازندگی عدد ۲ (مکان‌های دارای بیت یکسان) است.



(شکل-۵): محاسبه برازندگی

۴-۴-۲-۶جمعیت

مفهوم جمعیت[1] در الگوریتم رمزنگاری، جمعیتی از کلیدهای رمز است. برای مسأله کلیدهایی وجود دارند که می‌توانند به‌عنوان کلید، چه درست، چه غلط در نظر گرفته شوند که به این کلیدها، کلیدهای ممکن یا شدنی گفته می‌شود. تعداد جمعیت اولیه می‌تواند براساس هر حوزه کاری متفاوت باشد. در این پژوهش براساس آزمون‌های انجام‌شده، جمعیت اولیه ۱۰۰ کلید در نظر گرفته شده است. در هر نسل، جمعیت کلیدهای رمز با استفاده از پارامترهایی طراحی شده و تابع برازندگی به‌سوی بهینگی و تکامل پیش می‌روند و این فرایند تا سی نسل ادامه می‌یابد. کاهش زمان و کشف کلیدرمز، دو معیار در انتخاب جمعیت کلیدها (۱۰۰) و تکرار نسل (۳۰) بوده است که از نتایج آزمون‌های مختلف به‌دست آمده است.

۵- طراحی آزمایش‌ها

در این پژوهش، سامانه‌های مورد استفاده Core 2 Duo با پردازنده ۲,۵۳ گیگاهرتز و حافظه اصلی چهار گیگابایت جهت انجام آزمایش‌ها مورد استفاده قرار گرفته است. سامانه‌های مورد استفاده در کارهای پیشین که جهت انجام آزمایش‌ها استفاده شده Core 2 Duo بوده است [۴و۲،۳]. از بلوک‌های متنی مختلف جهت تحلیل رمز الگوریتم رمزنگاری SDES استفاده شده است و در تابع برازندگی از خصوصیات آماری تک‌حرفی، دوحرفی و حتی سه‌حرفی استفاده شده است. در کارهای پیشین از خصوصیات آماری تک‌حرفی استفاده شده است [۴و۲،۳]. در تحلیل‌رمز الگوریتم رمزنگاری SDES با توجه به ماهیت الگوریتم‌های فرااکتشافی، میانگینی از اجراهای مختلف (۱۰ اجرا در هر بلوک) به‌عنوان نتایج آزمایش‌ها در نظر گرفته شده است. طراحی ابتکاری پارامترها در الگوریتم ژنتیک جهت کشف کلید رمز الگوریتم رمزنگاری SDES در جدول (۲) آمده که براساس آزمون‌های مختلف صورت گرفته است.

(جدول-۲): تنظیم و طراحی ابتکاری پارامترها

پارامترها	مقدار
جمعیت اولیه	۱۰۰
انتخاب	نخبه‌گرایی (بهترین کلید)
برش	ستاره‌ای
جهش	سه نقطه ای- تصادفی
تعداد نسل	۳۰

پارامترهایی که برای الگوریتم ژنتیک پیشنهادی در نظر گرفته شده، بهترین مقادیری است که آزمون شده است.

۶- نتیجه و بحث

الگوریتم رمزنگاری SDES به‌عنوان یک الگوریتم رمزنگاری مدرن محسوب می‌شود که از معماری فیستلی استفاده کرده و سامانهٔ رمزنگاری آن از کلید متقارن استفاده کرده است که از یک کلید در عملیات رمزنگاری و رمزگشایی استفاده می‌شود. هدف در این پژوهش، بررسی عملکرد الگوریتم‌های جستجوی حمله فراگیر و فرااکتشافی در تحلیل رمز الگوریتم رمزنگاری SDES بوده است. الگوریتم جستجوی حمله فراگیر به‌عنوان یک الگوریتم جستجوی کامل، الگوریتم ژنتیک به‌عنوان یک الگوریتم هوش تکاملی و الگوریتم بهینه‌سازی توده ذرات به‌عنوان یک الگوریتم هوش جمعی در آزمایش‌ها استفاده شده و علاوه‌بر این الگوریتم‌ها، یک الگوریتم ژنتیک پیشنهادی نیز معرفی شده است که عملکرد این چهار الگوریتم روی الگوریتم رمزنگاری SDES مورد تحلیل و بررسی قرار می‌گیرد. بلوک‌های متنی متفاوتی در هر چهار الگوریتم (الگوریتم جستجوی حمله فراگیر، الگوریتم ژنتیک ، الگوریتم بهینه‌سازی توده ذرات دودویی و الگوریتم ژنتیک پیشنهادی) آزمون شده است. تحلیل رمز در دو معیار زمان کشف بیت‌های کلید رمز به ثانیه و کشف تعداد بیت‌های کلیدرمز در بلوک‌های متنی بررسی شده است. نتایج آماری کشف بیت‌های کلیدرمز در جدول (۳) آمده است.

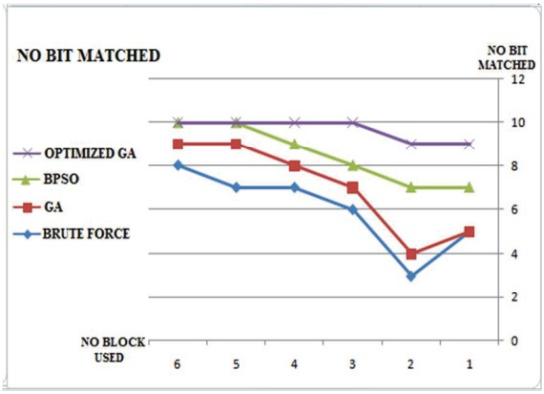
[1] Population

(جدول-۳):کشف بیت‌های کلیدرمز

بلوک متنی	تعداد حروف در بلوک متنی	الگوریتم جستجو حمله فراگیر (تعداد بیت‌های کشف شده)	الگوریتم ژنتیک (تعداد بیت‌های کشف شده)	الگوریتم بهینه‌سازی توده ذرات دودویی (تعداد بیت های کشف شده)	الگوریتم ژنتیک پیشنهادی (تعداد بیت‌های کشف شده)
۱	۲۰۰	۵	۵	۷	۹
۲	۴۰۰	۳	۴	۷	۹
۳	۶۰۰	۶	۷	۸	۱۰
۴	۸۰۰	۷	۸	۹	۱۰
۵	۱۰۰۰	۷	۹	۱۰	۱۰
۶	۱۲۰۰	۸	۹	۱۰	۱۰

همان‌طور که در جدول (۳) مشاهده می‌شود، بلوک‌های متنی متفاوتی مورد آزمون قرار گرفته است. در این جدول معیار تحلیل در هر بلوک متنی، تعداد بیت‌های کشف شده کلید رمز در الگوریتم رمزنگاری SDES است. در نتایج جدول به‌وضوح دیده می‌شود که در تمامی بلوک‌های متنی الگوریتم‌های فرااکتشافی در مقایسه با الگوریتم جستجوی حمله فراگیر تعداد بیت‌های بیشتری از کلید رمز را کشف کرده‌اند. در تحلیل الگوریتم‌های فرااکتشافی، الگوریتم بهینه‌سازی توده ذرات دودویی در مقایسه با الگوریتم ژنتیک کارهای پیشین عملکرد بهتری داشته است [۴و۲،۳]. در این پژوهش، الگوریتم ژنتیک پیشنهادی با تنظیم و باز طراحی پارامترها و طراحی الگوریتمی جهت کشف کلید رمز، دقت و هم‌گرایی سریع‌تری را در نتایج این الگوریتم به همراه داشته است به‌طوری‌که در مقایسه با الگوریتم‌های جستجو حمله فراگیر و الگوریتم ژنتیک [۳،۴]. دقت و هم‌گرایی سریع‌تر و در مقایسه با الگوریتم بهینه‌سازی توده ذرات دودویی [۲]. هم‌گرایی سریع‌تری در کشف بیت‌های کلیدرمز از خود نشان داده است. نمودار تعداد بیت‌های کشف‌شده در کلید رمز الگوریتم SDES در شکل (۶) آمده است.

در شکل (۶) الگوریتم جستجوی حمله فراگیر، الگوریتم ژنتیک، الگوریتم بهینه‌سازی توده ذرات دودویی، الگوریتم ژنتیک پیشنهادی استفاده شده است. محور افقی تعداد بلوک متنی مورد استفاده و محور عمودی تعداد بیت‌های کشف‌شده را در کلید رمز نشان می‌دهد که الگوریتم‌های فرااکتشافی در بلوک‌های مختلف متنی در مقایسه با الگوریتم حمله فراگیر عملکرد بهتری از خود نشان داده‌اند. نتایج آماری زمان کشف بیت‌های کلیدرمز در جدول (۴) آمده است.

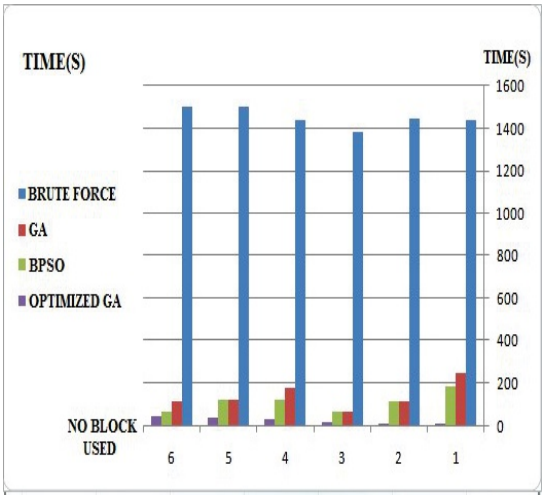


(شکل-۶): کشف بیت‌های کلیدرمز

(جدول-۴): زمان کشف بیت‌های کلیدرمز

بلوک متنی	تعداد حروف در بلوک متنی	الگوریتم جستجو حمله فراگیر (ثانیه)	الگوریتم ژنتیک (ثانیه)	الگوریتم بهینه‌سازی توده ذرات دودویی (ثانیه)	الگوریتم ژنتیک پیشنهادی (ثانیه)
۱	۲۰۰	۱۴۴۳	۲۴۷	۱۸۹	۳
۲	۴۰۰	۱۴۴۷	۱۲۱	۱۲۱	۸
۳	۶۰۰	۱۳۸۶	۶۹	۶۷	۱۲
۴	۸۰۰	۱۴۴۱	۱۸۱	۱۲۸	۱۶
۵	۱۰۰۰	۱۵۰۱	۱۲۶	۱۲۳	۲۰
۶	۱۲۰۰	۱۵۰۵	۱۲۱	۶۹	۲۵

همان‌طور که در جدول (۴) مشاهده می‌شود، بلوک‌های متنی متفاوتی مورد آزمون قرار گرفته است. در این جدول معیار تحلیل در هر بلوک متنی، زمان کشف بیت‌های کلید رمز در الگوریتم رمزنگاری SDES است. نتایج جدول به‌وضوح نشان می‌دهد که در تمامی بلوک‌های متنی زمان کشف بیت‌های کلید رمز با استفاده از الگوریتم جستجوی حمله فراگیر بیش از بیست دقیقه بوده است. در تحلیل الگوریتم‌های فرااکتشافی، در تمامی بلوک‌های متنی زمان کشف بیت‌های کلید رمز با استفاده از الگوریتم بهینه‌سازی توده ذرات دودویی و الگوریتم‌های ژنتیک کارهای پیشین بیش از یک دقیقه است [۴و۲،۳]. در تمامی بلوک‌های متنی زمان کشف بیت‌های کلید رمز با استفاده از الگوریتم ژنتیک پیشنهادی نیز کمتر از سی ثانیه است. در معیار زمان کشف بیت‌های کلید رمز الگوریتم رمزنگاری SDES، الگوریتم‌های فرااکتشافی عملکرد بهتری در مقایسه با الگوریتم جستجوی حمله فراگیر از خود نشان داده‌اند. نمودار زمان کشف بیت‌های کلید رمز در شکل (۷) آمده است.



(شکل-۷): زمان کشف بیت‌های کلیدرمز

در شکل (۷) الگوریتم جستجوی حمله فراگیر، الگوریتم ژنتیک، الگوریتم بهینه‌سازی توده ذرات دودویی، الگوریتم ژنتیک پیشنهادی استفاده شده است. محور افقی تعداد بلوک مورد استفاده و محور عمودی زمان کشف بیت‌های کلید رمز را نشان می‌دهد که با مشاهده نمودار، کاهش زمان محسوسی در الگوریتم‌های فرااکتشافی در مقایسه با الگوریتم حمله فراگیر دیده می‌شود.

۷- نتیجه‌گیری

در این پژوهش، جهت تحلیل‌رمز الگوریتم رمزنگاری SDES از الگوریتم جستجوی حمله فراگیر و الگوریتم‌های فرااکتشافی استفاده شده است. در انتخاب الگوریتم فرااکتشافی از الگوریتم ژنتیک به‌عنوان الگوریتم هوش تکاملی و از الگوریتم بهینه‌سازی توده ذرات به‌عنوان الگوریتم هوش جمعی استفاده شده است علاوه بر این الگوریتم‌ها، یک الگوریتم ژنتیک پیشنهادی با تنظیم و طراحی ابتکاری پارامترها و طراحی الگوریتمی جهت کشف کلید رمز معرفی شد. نتایج نشان می‌دهند که الگوریتم‌های فرااکتشافی در مقایسه با الگوریتم جستجوی حمله فراگیر در معیارهای کشف بیت‌های کلیدرمز و زمان کشف بیت‌های کلیدرمز عملکرد بهتری از خود نشان داده‌اند. در تحلیل الگوریتم‌های فرااکتشافی نیز الگوریتم ژنتیک پیشنهادی در مقایسه با الگوریتم بهینه‌سازی توده ذرات دودویی و الگوریتم‌های ژنتیک کارهای پیشین عملکرد بهتری از خود نشان داده است. با استفاده از الگوریتم‌های فرااکتشافی مناسب در آینده بلوک‌های پیچیده‌ای چون DES ، 3DES و AES مورد تحلیل رمز قرار خواهد گرفت.

۸- مراجع

[۱] ملکیان، ا.، ذاکرالحسینی، ع. "امنیت داده‌ها". ویرایش پنجم، تهران، انتشارات علمی فرهنگی نص، ۱۳۹۵.

[2] L.Sharma, B.k. Pathak, N.Sharma, "Brea-king of Simplified Data Encryption Standard Using BinaryParticle Swarm Optimization," *International Journal of Computer Science Issues*, vol.9, pp.307-313,2012.

[3] P.Garg, Sh.Varshney, M.Bhardwaj, " Cryptanalysis Of Simplified Data Encryption Stan-dard Using Genetic Algorithm," *American Journal Of Networks And Communications*, vol.4, pp. 32-36, 2015.

[4]L.Sharma, B.K. Pathak, R.Sharma, "Breaking of Simplified Data Encryption Standard Using Genetic Algorithm," *Global Journal of Computer Science and Technology*, vol.12, pp.12, 55-60, 2012.

[5] M.M. Alani, "DES96 - Improved DES Security". *7th International Multi-Conference on IEEE*, Gulf: Bahrain, 2010, pp. 27-30.

[6] Kh.Alallayah, M. Amin, W.Abdelwahed, A. Alhamami, "Applying Neural Network For Simplified Data Encryption Standard (SDES) Cipher System Cryptanalysis," *The International Arab Journal Of Information Technology*, vol. 9, PP.163-169, 2012.

[7] M.M. Alani, "Neuro-Cryptanalysis of DES," *Internet Security (WorldCIS) on IEEE*, Muscat, Oman, 2012, pp.10-12.

[8] T.Mekhaznia, A.Zidani, M. Derdour, " A New Approach Of Known Plaintext Attack With Genetic Algorithm," *Wseas Transactions On Computers*, 2018, vol.17, pp.18–32.

[9] R.Vimalathithan, M.L.Valarmathi, "Cryptana-lysis of DES using Computational Intelligence," *European Journal of Scientific Research*, vol.55, pp. 237-244, 2011.

[10] V.Saroha, S.Mor, J.Malik, "A Review of Various Techniques of Cryptanalysis," *Inter-national Journal of Advanced Research in Computer Science and Software Engineering*, vol.2, PP.89-92, 2012.

[11] G.Shrivastava, "Analysis Improved Cryptosystem Using DES with RSA," *VSRD International Journal of Computer Science & Information Technology*, vol. 1, pp.465-470, 2011.

[12] P.A. Bagane, S.Kotrappa, "Cryptanalysis For S-DES Using Genetic Algorithm," *International Conference On Smart Electronic Systems*, 2016,Vol.3, pp. 42–44.

[13] N.S. Habib, R. Awan,W. Haider, "A Modified Simplified Data Encryption Standard Algorithm,

" *International Journal Of Computer Science And Software Engineering*, vol.6, PP.152-154, 2017.

[14] C.R.Gopal, P.Rajkumar, "Optimized Approach For Secure Communication Using Des Algorithm, " *International Journal of Pure and Applied Mathematics*, vol.116, pp.125-130, 2017.

[15] H. Zodpe, P. Wani, R.Mehta, "Hardware Implementation Of Algorithm For Cryptanalysis," International Journal On Cryptography And Information Security, vol.3, PP.7-15,2013.

[16] D.U.Jeswani, S.G. Kale, "The Particle Swarm Optimization Based Linear Cryptanalysis Of Advanced Encryption Standard Algorithm, " *International Journal On Recent And Innovation Trends In Computing And Communication*, vol. 3, PP.1767-1769,2015.

[17] A.Bhateja, "Analysis Of Different Cryptosystems Using Meta-Heuristic Techniques," I*EEE International Conference On Advanced Communication Control And Computing Technologies*, 2014, pp. 1931–1934.

[18] K. Salabat, S.Waseem, A.K. Farrukh, "Cryptanalysis of Four-Rounded DES using AntColony Optimization," *Information Science and Applications (ICISA) On IEEE*, pp.21-23, 2010.

[19] B. Akiwate, V. Desai, "Artificial Neural Networks for Cryptanalysis of DES," *International Journal of Innovations in Engineering and Technology*, vol.2, pp.11-17, 2013.

[20] P. Nema, A. Jain, "A Comparative Survey on Various Encryption Techniques for Information Security," I*nternational Journal of Advanced Research in Computer Science and Software Engineering*, vol.3, pp.725-730, 2013.

[21] T.Tadros, A.E.F. Hegazy, A. Badr, "Genetic Algorithm for DES Cryptanalysis," *International Journal of Computer Science and Network Security*, vol.10, pp.5-11, 2010.

[22] A.Dadhich, S.K.Yadav, "Swarm Intelligence And Evolutionary Computation Based Cryptography And Cryptanalysis Of 4-Round Des Algorithm, " *International Journal Of Advanced Research In Computer Engineering & Technology*, vol. 3, pp.1624-1633, 2014.

[23] Rajashekarappa, K.M.S. Soyjaudah, "Comparative Cryptanalysis Of Simplified- Data Encryption Standard Using Tabu Search And Simulated Annealing Methods ," *International Journal Of Engineering Research And Development*, vol. 5, pp.7-12, 2012.

[24] M.Moradi, H. Khotanlou, M. Abbasi, "Breaking Of Simplified-Data Encryption Stan-dard Using Optimized SPSO," *The Scientific Journal Of Advanced Defence Science And Technology*, vol.3, pp.203-210, 2014.

[25] S. Zoubir, A. Tragha, "Uses Of Genetic Algorithm In Cryptanalysis Of RSA," *IOSR Journal Of Computer Engineering*, vol. 17, pp.465-470, 2016.

[26] Adwan, S.A,Shraideh, M.A,Saleem, M.R. Al Saidat, " A Genetic Algorithm Approach For Breaking Of Simplified Data Encryption Standard," *International Journal Of Security And Its Applications, 2015*, vol. 9, pp.295-304, 2015.

[27] A.K.Kendhe, H. Agrawal, "A Survey Report On Various Cryptanalysis Techniques," *International Journal Of Soft Computing And Engineering*, vol. 3, pp.287-293,2013.

میثم مرادی تحصیلات خود را در مقاطع تحصیلی کاردانی و کارشناسی در رشته مهندسی کامپیوتر گرایش نرم‌افزار به‌ترتیب در سال ۱۳۸۴ در دانشگاه فنی بروجرد و سال ۱۳۸۶ در دانشگاه آزاد اسلامی واحد ملایر به پایان رساند و مقطع کارشناسی ارشد خود را در سال ۱۳۹۲ در دانشگاه علوم تحقیقات تهران (همدان) در رشته مهندسی کامپیوتر گرایش نرم‌افزار به پایان رساند او هم‌اکنون در دانشگاه ملایر تدریس می‌کند. از زمینه‌های مورد علاقه وی می‌توان به امنیت داده‌ها، امنیت شبکه‌های رایانه‌ای، مهندسی نرم‌افزار، اختراع و نوآوری اشاره کرد.

مهدی عباسی تحصیلات خود را در سال ۱۳۸۰ در مقطع کارشناسی مهندسی کامپیوتر- سخت افزار و همچنین در سال ۱۳۸۵ در مقطع کارشناسی ارشد معماری سامانه‌های کامپیوتری در دانشگاه صنعتی شریف به پایان رساند. در سال ۱۳۹۱ در مقطع دکترای رشته معماری سامانه‌های رایانه‌ای، از دانشگاه اصفهان فارغ‌التحصیل شد. او از سال ۱۳۹۱ استادیار گروه کامپیوتر دانشگاه بوعلی سینا بوده و در حوزه‌های پردازنده‌های شبکه‌ای، پردازش سیگنال و طراحی مدارهای خیلی فشرده پژوهش می‌کند.